ISSN 1561-5359. Штучний інтелект, 2019, № 3-4

УДК 004.93

С.С. Кондратюк
Київський національний університет імені Тараса Шевченка, Україна
вул. Володимирська, 60, м. Київ, 01601

РОЗПІЗНАВАННЯ ЖЕСТІВ УКРАЇНСЬКОЇ ДАКТИЛЬНОЇ АБЕТКИ ЗА ДОПОМОГОЮ КРОСПЛАТФОРМНИХ ТЕХНОЛОГІЙ ТА ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ

S. Kondratiuk
Taras Shevchenko National University of Kyiv, Ukraine
60, Volodymyrska St., Kyiv, 01601

UKRAINIAN DACTYL ALPHABET GESTURE RECOGNITION USING CROSS PLATFORM SOFTWARE AND CONVOLUTIONAL NEURAL NETWORKS

Запропоновано технологію, розроблену за допомогою кросплатформних засобів, для моделювання жестів української дактильної абетки, анімації переходів між станами жестових одиниць та комбінування жестів (слів). Розроблена технологія відтворює послідовність жестів за допомогою віртуальної просторової моделі руки та виконує розпізнавання дактилем із вхідного потоку камери за допомогою навчених на зібраному наборі зображень згорткової нейронної мережі. За допомогою кросплатформних засобів технологія може бути запущеною на багатьох платформах без необхідності портування або розробленою заново під кожну платформу окремо.

Ключові слова: кросплатформність, мова жестів, моделювання дактилем, розпізнавання дактилем, згорткові нейронні мережі

The technology, which is implemented with cross platform tools, is proposed for modeling of gesture units of sign language, animation between states of gesture units with a combination of gestures (words). Implemented technology simulates sequence of gestures using virtual spatial hand model and performs recognition of dactyl items from camera input using trained on collected training dataset set convolutional neural network. With the cross platform means technology achieves the ability to run on multiple platforms without re-implementing for each platform.

Keywords: cross platform, sing language, dactyl modeling, dactyl recognition, convolutional neural networks

Introduction

Sign language is one of major means for information transition, alongside with text and speech. Sings can be used to define specific letters, words, phrases and can be processed, encoded and stored in a various ways. Developing a technology for persisting and modeling signs and sign languages is a challenging problem due to differences in available platforms. Different platform have different operating systems (such as mobile - iOS, Android, desktop - MacOS, Linux, Windows, and web - ChromeOS, etc), which implies different performance level and requires porting the codebase on each platform; some platform require internet connection (such as cloud computing technologies [1] or we operating systems) and others do not, etc. Presenting such a technology for sing language is an actual problem for people with hearing disabilities and their relatives, but also is important in a wider appliance, due to universality of sing language.

Cross-platform technologies [2] provide a way to overcome this problem. Cross-platform technologies can be used instead of virtual-machines [3] or a set of mono-platform technologies. Using these technologies allows to develop a single codebase for different type of platforms, independent of CPU type, operating system of hardware performance and to deploy it on all platform seamlessly.

In this article a solution for the problem of sing language modeling and recognition is proposed based on cross-platform technologies. The sign language modeling and recognition performance can be flexible and adjusted, based on the hardware it operates or based on availability of internet connection. The proposed approach tunes the complexity of the 3D hand model (parameters such as the number of polygons for rendering the hand and the animation step of sings transition) based on the CPU type, amount of available memory and internet connection speed. The sing recognition is also performed using cross-platform technologies and can be adjust the tradeoff in model size and performance. The sing (gesture) modeling and recognition is a part of a single gesture communication technology and this paper is a further development of author's previous works [4], [5].

Existing approaches for modeling and recognition of sign language and their implementtation on different platforms

A technology for both sing language modeling and recognition can be considered as a set of separate dependent or independent technologies for gesture modeling and gesture recognition. Some of systems provide only of the stated technologies, typically only for a specific platform. American Sing Language Online Dictionary [6] is one of the first systems which were assembled for gesture modeling, however it is based on a set of recored videos, stored in a database, and this approach was applied in a set of agencies [7]. However, due to its method of persisting gestures and no possible ability to modify them, this system is not flexible and limited only to a set of pre-recorded gestures. Also, no gesture recognition method is provided.

Three-dimensional hand model is an important part of gesture language modeling. Two groups of hand modeling presentation are researched in the work [8]. In the paper, spatial approach takes into account positioning of the hand sign and their parametrical representation. The temporal approach is based on the rules of transitions of a gesture.

[9] develops a technology for modeling gesture for an input text. The technology consist of a statistical model for given text processing and a generative algorithms for appropriate hand gesture modeling, using specified kinematics. As a result of the work, authors provide ANVIL tools for annotation, DANCE library for sing transition and sing generator NOVA [10]. However, the technology is specified to work only on Windows operating system and x86 CPU. Similar technology for modeling symbols is proposed in [11], however also only for a single platform. Similar technology for smartphones is proposed in [13], however the approach consists only of gesture recognition.

Problem statement

The proposed technology should consist of two parts, which are sign language [11], [12] modeling and gesture recognition module. Both modules should be able to run without codebase modification on multiple platforms and should be developed using cross-platform tools.

Gesture recognition module should consist of a model which is able to detect and identify the gesture, specified by the user, from a camera input. Set of gestures is limited by the Ukrainian dactyl language, but can be extended further. An appropriate dataset of Ukrainian dactyl language should be collected for testing the model performance. The sing language modeling module should be able to reproduce a gesture specified by a set of parameters, stored in a database, and should be limited by a set of Ukrainian dactyl language signs, but can be extended further with other languages. The gesture modeling module should also be able to reproduce gesture transitions, meaning it can model seamlessly words and sentences, consisting of Ukrainian dactyl language signs.

Proposed approach

To developed a technology for Ukrainian dactyl language modeling and recognition, which can run on multiple platforms, without changing the codebase, an approach based on cross-platform tools is proposed. Gesture modeling module should consist of a virtual three dimensional hand model and a user interface, which should provide the user with ability to specify a symbol or a set of symbols, which then will be transitioned as a sequence of gestures. To implement both hand model and user interface, a cross-platform framework Unity3D [14] was used. Comparing to other 3D engines, it provides a unified development process for all available platforms (mobile, desktop and web) and provides a seamless way to deploy the application on all of them without changing the codebase. To develop a gesture recognition module, a cross-platform framework Tensorflow [15] is proposed. This approach based on cross-platform framework for machine learning allows to developed and train a gesture recognition model once, and then deploy it on multiple platforms (mobile, desktop and web) without any modifications to the model or the code for training. As a model architecture, the MobileNet architecture is considered, enhanced with 3D convolutions, to take into account temporal information from a sequence of input frames from the camera. Altogether, the proposed technology novelty is that it's a unified cross-platform technology for Ukrainian dactyl language modeling and recognition, with improved MobileNet architecture for improved recognition of the Ukrainian dactyl alphabet.

Infologic model

The technology architecture diagram (Figure 1) demonstrates the interaction of main components of the proposed system. The gestures for gesture modeling are stored in a specified format (YAML [16]) in a database, and are exploited by the gesture modeling engine for setting a configuration of a spatial three-dimensional hand model using specified parameters for the gesture from a corresponding database entry. The gesture modeling module operates over the gesture database and is a part of the application, which consists of gesture modeling and UI components, both of them being developed with Unity3D framework, using C# programing language. The virtual hand model is specified by a skeleton and a set of parameters and their limitations for each skeleton joint. The gesture recognition module is implemented with Tensorflow framework, using Python programing language. The gesture recognition module run independently of gesture modeling module and uses gestures database. Main components of the gesture recognition module is the model which performs gesture recognition, and the wrapper which converts camera input into appropriate format for the model.

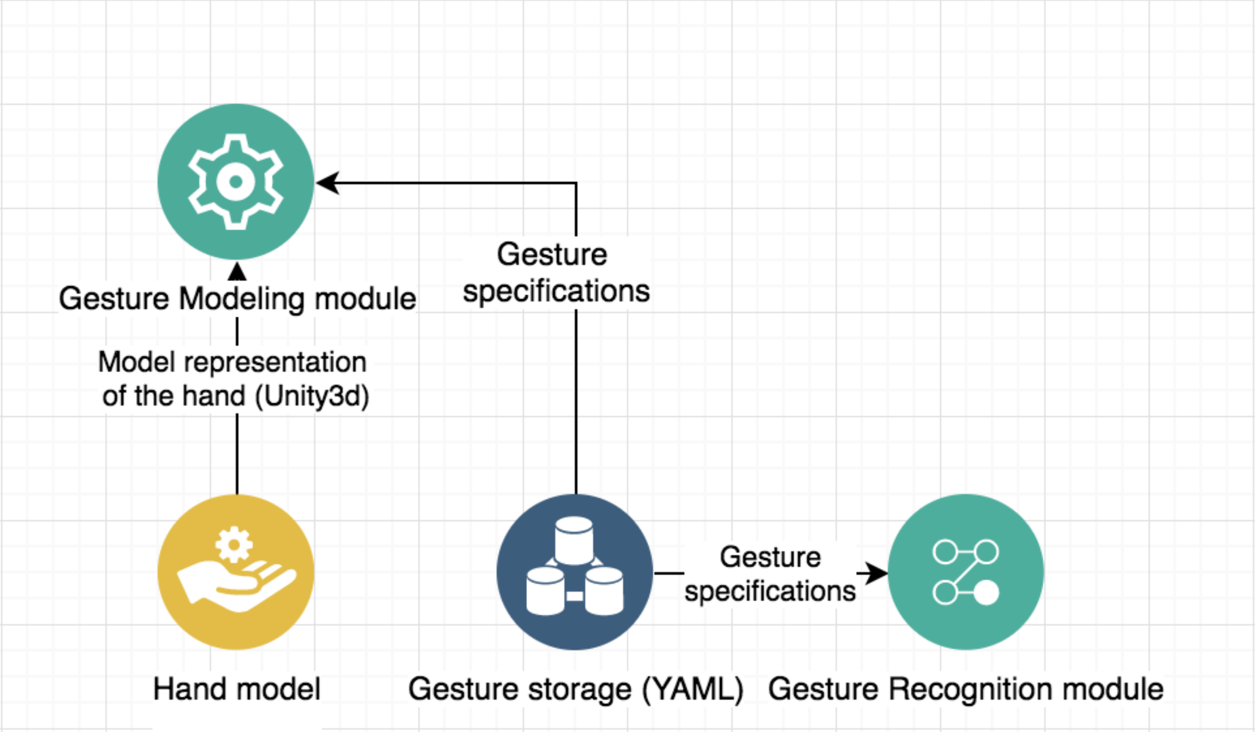


Fig. 1. Infologic model of cross platform gesture communication technology.

Gesture modeling

The three-dimensional hand model skeleton was implemented with real human hand anatomy in mind. Skeleton model consists of: 8 bones in wrist, 3 bones in the thumb and 1 metacarpus and 3 phalanges in each of the other fingers. Each joint of each pair of bones has it's own type of connection and it's own parameters for setting this joint, it's own degree of freedom and it's limitations. Overall, the hand model is represented with a skeleton which consists of 27 bones and has 25 degrees of mobility. The thumb has 5 degrees of freedom, middle and index fingers have four degrees of freedom, four degrees of freedom are located in the metacarpal-carpal joint to the little finger and thumb to enable movement of the palm.

Unity3D framework was used for implementing the three dimensional hand model, since developing your own cross-platform rendering engine is a non-trivial task. Unity3D was selected due to friendly user interface, ability to implement through it's means both the scene and user interface. Over the hand skeleton, a realistic hand model was developped, rendered with more than 70,000 polygons (Figure 2), Unity3D framework is able to handle such model with satisfactory performance.

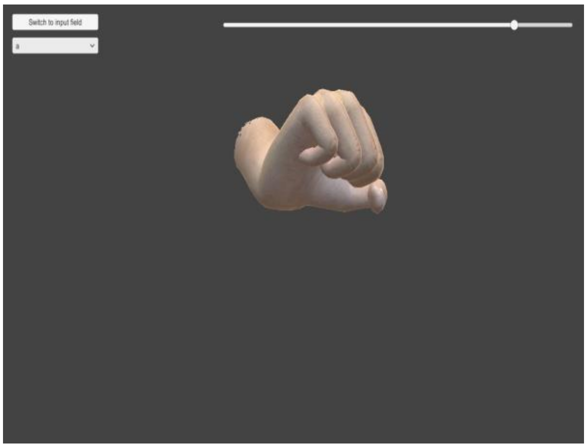
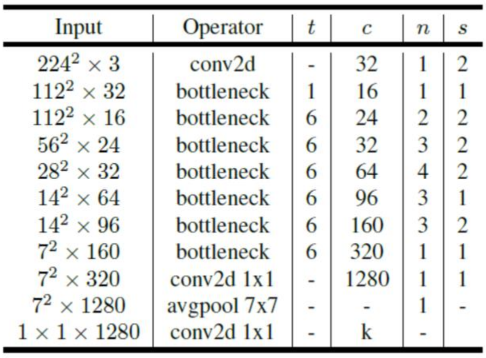


Fig. 2. Gesture modeling under
iOS platform.

Gesture recognition

Gesture recognition, as a part of cross-platform technology for Ukrainian dactyl language modeling and recognition, should be implemented using cross-platform tools. Gesture recognition approach depend on the type of input information they work with. In case of 3D model bases algorithms or skeletal-based algorithms, the approach can use volumetric or skeletal model, or a combination of them. Although, these approaches tend to be computationally expensive and require additional hardware from user. Other type of approaches, appearance-based models derive parameters directly from the image or a sequence of images (in case video is used as an input). As a next step some pattern mining technique or machine learning approach is used to train a recognition model. Due to no need in additional hardware apart from a simple webcamera, these type of approaches were selected for the cross-platform technology. Some approaches, for example, Ong et al. [17] proposes Sequential Pattern Mining in order to detect signs based on the tree structures.

Convolutional Neural Networks (CNN) is a class of deep neural networks which are regularized versions of multilayer perceptrons, most commonly applied to analyzing images and videos. CNNs are especially good at analyzing images due to ability to take into account locality reference of the data in the image (typically nearby samples at some input data are not related, which is not true in case of an image). Therefore, CNN show state-of-the-art results in image classification and recognition tasks [18], [19]. Another benefit of the convolutional neural networks is no need in hand-crafted features, unlike conventional pattern matching algorithms. The process of training takes the input data and finds all the features needed for recognition and stores them as weights of the model. CNNs are robust at the task of classification or recognition of the object on an image, independent of input image scale, lightning conditions, occlusions, noise, etc. Although training such a model requires a sufficient dataset. Typically architecture of the CNN consists of a set of convolutional, pooling and ReLU layers. Tensorflow framework provides a cross-platform and performance-efficient implementation of convolutional neural networks.



Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

Fig. 3. Architecture of MobileNetv2.

Gesture recognition experiment

Training a CNN model requires an appropriate dataset. The data should not be biased, therefore for the task of Ukrainian dactyl alphabet gestures recognition the dataset should be diverse. Since the temporal information also provides useful information about gestures transitions, each sample of the gestures should exist as a sequence of frames or a short video, which is also increasing the dataset. For the gesture recognition experiment a dataset of 50,000 images was collected, with 50 different hands recorded, with almost 1,000 images per each person. Distribution of male hands was 55% and female 45%.

The dataset was collected using differrent surroundings. The background was changing for each person after each 50 frames recorded. Different lighting conditions were setup during recording (20%/30%/50% in low, medium, bright light conditions correspondingly). 20% of the dataset was distorted with noise and blur with gaussian noise and gaussian blurring.

MobileNetV2 [20] architecture (Figure 3) is a new mobile architecture, development of the MobileNet model. MobileNetV2 extends its predecessor with 2 main ideas. Residual blocks connect the beginning and end of a convoluteonal block with a skip connection. By adding these two states the network has the opportunity of accessing earlier activations that weren't modified in the convolutional block. This approach turned out to be essential in order to build networks of great depth. On the other hand, MobileNetV2 follows a narrow->wide->narrow approach. The first step widens the network using a 1x1 convolution because the following 3x3 depthwise convolution already greatly reduces the number of parameters. Afterwards another 1x1 convolution squeezes the network in order to match the initial number of channels. In the table (Figure 3) it is how the bottleneck blocks are arranged. t stands for expansion rate of the channels. A factor of 6 opposed to the 4 in our example. c represents the number of input channels and n how often the block is repeated. Lastly s tells whether the first repetition of a block used a stride of 2 for the downsampling process. This is a common assembly of convolutional blocks.

Process of training MobileNetv2 network for gesture detection takes ~ 200.000 iterations. Figure 4 shows curves of how the neural network is optimized. Such a training allowed to achieve accuracy of 98% on testing dataset, of additional size of 15,000 images, with 30 different hands recorded, with almost 500 images per each person.

Figure 5 demonstrates user interface of gesture recognition module. Letter B of Ukrainian dactyl alphabet is detected and shown with a bounding box.

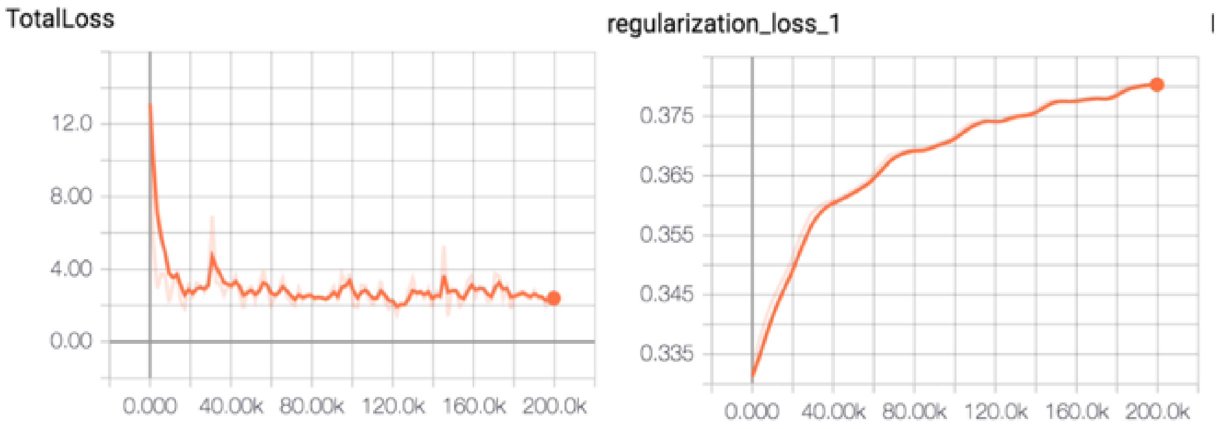


Fig. 4. Loss and regularization loss curves of MobileNet training process.

Application of cross platform tools

The proposed technology is implemented using cross platform tools. The gesture modeling and recognition modules are both developed using cross platform frameworks, which operate over a databased with gestures in a unified format (YAML). To overcome the problem of running the technology with satisfying performance on multiple platform, settings for performance adjustment were developed. Based on the platform and its initial hardware analysis, the dimensional hand model is adjusted. Number of polygons can be lowered, and step of performance can be decreased.

However, there are no specific hardware requirements for the technology to run, as the used cross-platform frameworks (Tensorflow, Unity3D) don't require.

If the available hardware does not meet the minimum requirements of information technology, the user is given the recommendation to choose “online” mode, in which the calculation is not performed on hardware.

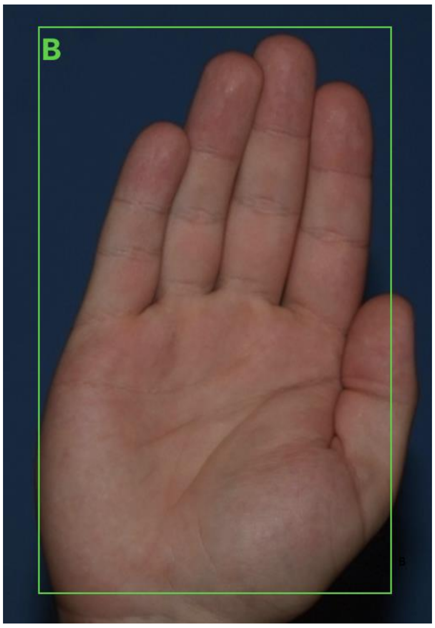


Fig. 5. Example of recognition UI.

Conclusions

The proposed technology consists of two main modules: gesture modeling and gesture recognition modules, which use the database with gestures specifications stored in YAML format in a PostgreSQL [21] database.

The proposed technology implements gesture modeling and gesture recognition for Ukrainian dactyl alphabet gestures. A dataset of 50.000 images was collected using diverse conditions and different persons hands. Gesture modeling was implemented using Unity3D framework, which is cross-platform and shows satisfying performance on different platforms (mobile, web and desktop) while rendering a realistic three dimensional hand model. Number of polygons and animation step of gesture transitions can be adjusted for the sake of performance. Gesture recognition module was implemented using Tensorflow framework, which provides ability to deploy its model on different platforms without any codebase modifications. As a model for gesture recognition, MobileNetv2 architecture was chosen, as a model with best trade-off of size and accuracy, especially on low performance platforms (such as mobile and web). The model was trained on the collected Ukrainian dactyl language dataset.

The proposed gesture communication technology can be further augmented with other gestures and languages, and with other cross-platform modules.

References

1. Peter Mell and Timothy Grance (September 2011). The NIST Definition of Cloud Computing (Technical report). National Institute of Standards and Technology: U.S. Department of Commerce. doi:10.6028/NIST.SP.800-145. Special publication 800-145.
2. The Linux Information Project, Cross-platform Definition.
3. Smith, James; Nair, Ravi (2005). “The Architecture of Virtual Machines”. Computer. IEEE Computer Society. 38 (5): 32–38.
4. I. Krak, S. Kondratiuk (2017). Cross-platform software for the development of sign communication system: Dactyl language modelling, Proceedings of the 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2017, 1, pp. 167-170. DOI: 10.1109/STC-CSIT.2017.8098760
5. Yu.V. Krak, Yu.V. Barchukova, B.A. Trotsenko (2011). Human hand motion parametrization for dactylemes modeling, Journal of Automation and Information Sciences, 43 (12), pp. 1-11.
6. ASL Sing language dictionary [http://www.signasl.org/sign/model]
7. Apple Touchless Gesture System for iDevices [http://www.patentlyapple.com/patently-apple/2014/12/apple-invents-a-highly-advanced-air-gesturing-system-for-future-idevices-and-beyond.html]
8. Comparative study of hand gesture recognition system, Rafiqul Zaman Khan, Noor Adnan Ibraheem, Natarajan Meghanathan, et al. (Eds): SIPM, FCST, ITCA, WSE, ACSIT, CS & IT 06, pp. 203–213, 2012.
9. Gesture Modeling and Animation by Imitation, Michael Neff, Michael Kipp, Irene Albrecht, Hans-Peter Seidel, MPI–I–2006–4-008 September 2006.
10. Dynamic Controller Toolkit, Ari Shapiro, Derek Chu, Brian Allen, Petros Faloutsos, 2005

[http://www.arishapiro. com/Sandbox07_DynamicToolkit.pdf]

11. Iu.G. Kryvonos, Iu.V. Krak, 2011: Modeling human hand movements, facial expressions, and articulation to synthesize and visualize gesture information, Cybernetics and Systems Analysis, 47 (4), pp. 501-505.
12. Iu.G. Kryvonos, Iu.V. Krak, O.V. Barmak, D.V. Shkilniuk, 2013: Construction and identification of elements of sign communication, Cybernetics and Systems Analysis, 49 (2), pp. 163-172.
13. Android based portable hand sign recognition system, Jagdish L. Raheja, March 2015. DOI: 10.15579/gcsr.vol3.ch1 · Source: arXiv
14. Unity3D framework [https://unity3d.com/]
15. Tensorflow framework documentation [https://www.tensorflow.org/api/]
16. YAML – The Official YAML Web Site [http://yaml.org/]
17. Eng-Jon Ong et al. Sign language recognition using sequential pattern trees. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE. 2012, pp. 2200–2207.
18. American Sign language: Real-time American Sign Language Recognition with Convolutional Neural Networks Brandon Garcia Stanford University Stanford, CA, 2015.
19. Hand gesture recognition using neural network based techniques, Vladislav Bobic, School of Electrical Engineering, University of Belgrade, 2016.
20. Andrew G. Howard, Menglong Zhu, Bo Chen. “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Application”, 2017.
21. PostgreSQL official web site [https://www.postgresql.org/]

РЕЗЮМЕ

С. С. Кондратюк

Розпізнавання жестів української дактильної абетки за допомогою кросплатформних технологій та згорткових нейронних мереж

Мова жестів є одним із основних засобів передачі інформації, поряд із текстом і мовою. Як правило, у кожної країни є своя рідна мова жестів, проте напевно невідомо, скільки мов жестів існує в усьому світі. Українська мова жестів та український алфавіт дактилем є одним із найпоширеніших засобів спілкування в Україні після текстового та розмовного спілкування.

Надання технології вивчення жестів (знаків, дактилем) української мови для такої спільноти є актуальною проблемою та складним завданням.

Для вирішення завдання моделювання мови жестів та виконання анімації жестових структур за допомогою просторової віртуальної моделі руки, пропонується кросплатформна технологія, заснована на кросплатформній бібліотеці Unity3D. Кросплатформна бібліотека Unity3D також використовується для інтерфейсу користувача, технологія реалізована за допомогою мови програмування C#. Запропоновані інструменти можуть вирішити проблему запуску технології на декількох існуючих платформах. Новизна запропонованої технології полягає в тому, що вона є кросплатформною та має настроюваний рівень полігонів для тривимірної моделі руки та крок-анімації для переходів жестів.

Модель руки, вбудована в модуль моделювання жестів, має 27 кісток, кожна кістка з'єднана з іншою через різні типи суглобів. У рамках технології моделювання реалізовано тривимірну модель руки та реалізовано анімації переходів між жестами (морфемами). Дана технологія здатна відтворити реалістичну модель руки, що складається з понад 70000 полігонів.

Модулі розпізнавання жестів, розроблені за допомогою кросплатформних інструментів (засновані на Python, C ++), можуть бути вбудовані в інформаційну технологію. Конволюційні нейронні мережі показали надійні результати в задачах з розпізнавання зображень та жестів. Для експерименту був зібраний набір даних з дактилемами української мови. Кожен жест складається з 1000 зразкових зображень. 50 різних людей показували жести: з розподілом 70% чоловічих та 30% жіночих рук. Були використані різні умови освітлення (з розподілом 20% зображень у поганих, 30% у посередніх та 50% у хороших умовах освітлення), 10% зображень були спотворені шумом та розмиттям.

Архітектура MobileNet була використана як основа для архітектури CNN, і навчання тривало ~ 200 000 ітерацій, що становить приблизно 10 епох, і досягнуто ~98% точності на тестувальному наборі даних.

Надійшла до редакції 03.04.2019